Rao-Blackwellised Particle Filtering for Fault Diagnosis

Nando de Freitas
Department of Computer Science
University of British Columbia
2366 Main Mall
Vancouver, B.C. V6T 1Z4, Canada
Phone: (604) 822-6770
Fax: (604) 822-5485
nando@cs.ubc.ca

Abstract— We tackle the fault diagnosis problem using conditionally Gaussian state space models and an efficient Monte Carlo method known as Rao-Blackwellised particle filtering. In this setting, there is one different linear-Gaussian state space model for each possible discrete state of operation. The task of diagnosis is to identify the discrete state of operation using the continuous measurements corrupted by Gaussian noise.

TABLE OF CONTENTS

1. INTRODUCTION

Automatic fault diagnosis is a fundamental problem in aerospace systems and autonomous robotics. Planetary rovers, for example, must be able to diagnose faults and carry out repairs without ground-operator intervention [18]. In a typical diagnosis scenario, the machine (*e.g.* electronic system or robot) receives a continuous stream of data from various on-board sensors. It, subsequently, processes this information to identify its discrete state of operation (*e.g.* "stuck rear wheel", "normal operation", "damaged camera").

In the design of an appropriate diagnosis system, we need to take into consideration that both the machine and its environment change with time, the measurements are corrupted by noise, some of the quantities of interest are unobservable, and the machine's state tends to be different under different operating conditions. To address these constraints, we need to adopt probabilistic dynamic models. These models describe the the evolution of the machine's discrete and continuous states and the statistical properties of the data. Typically, the continuous states and measurements are assumed to be Gaussian distributed.

One problem with the required models is that they are intractable. This has motivated the development of algorithms for approximate inference [18], [17]. The state-of-the-art method is the Monte Carlo particle filter proposed in [17]. This method allows one to compute, recursively in time, a stochastic point-mass approximation of the posterior distribution of the states given the observations. For a comprehensive review of Monte Carlo particle filtering methods see [7].

In this paper, we propose a substantially more efficient Monte Carlo particle filter for fault diagnosis. This algorithm exploits some of the analytical structure of the model. In particular, it exploits the fact that when we know the values of the discrete states, it is possible to compute the distribution of the continuous states exactly. We, therefore, combine a particle filter (PF) to compute the distribution of the discrete states with a bank of Kalman filters to compute the distribution of the continuous states. That is, we approximate the posterior distribution with a recursive, stochastic mixture of Gaussians. This strategy is known as *Rao-Blackwellisation*, because it is related to the Rao-Blackwell formula: see [4] for a general discussion. Rao-Blackwellised particle filters (RBPF) for mixtures of Gaussians are discussed in [1], [8].

2. MODEL AND INFERENCE OBJECTIVES

We adopt the following jump Markov linear Gaussian model:

$$\begin{aligned} z_t &\sim P(z_t|z_{t-1}) \\ x_t &= A(z_t)x_{t-1} + B(z_t)w_t + F(z_t)u_t \\ y_t &= C(z_t)x_t + D(z_t)v_t + G(z_t)u_t, \end{aligned}$$

where $y_t \in \mathbb{R}^{n_y}$ denotes the observations, $x_t \in \mathbb{R}^{n_x}$ denotes the unknown Gaussian states, $u_t \in \mathcal{U}$ is a known control signal, $z_t \in \{1, \ldots, n_z\}$ denotes the unknown discrete states. The noise processes are *i.i.d* Gaussian: $w_t \sim \mathcal{N}(0, I)$ and $v_t \sim \mathcal{N}(0, I)$. For clarity of presentation, we state explicitly that our model implies the con-

0-7803-7231-X/01/$10.00/© 2002 IEEE
IEEEAC paper # 493

tinuous densities

$$\begin{aligned} p(x_t|z_t, x_{t-1}) &= \mathcal{N}(A(z_t)x_{t-1} + F(z_t)u_t, B(z_t)B(z_t)^{\mathsf{T}}) \\ p(y_t|x_t, z_t) &= \mathcal{N}(C(z_t)x_t + G(z_t)u_t, D(z_t)D(z_t)^{\mathsf{T}}). \end{aligned}$$

The parameters $(A, B, C, D, E, F, P(z_t|z_{t-1}))$ are known matrices with $D(z_t)D(z_t)^{\mathsf{T}} > 0$ for any z_t; see for example [2] for parameter estimation and model selection. Finally, the initial states are $x_0 \sim \mathcal{N}(\mu_0, \Sigma_0)$ and $z_0 \sim P(z_0)$.

The important thing to notice is that for each realisation of z_t, we have a single linear-Gaussian model. If we knew z_t, we could solve for x_t exactly using the Kalman filter algorithm.

The aim of the analysis is to compute the marginal posterior distribution[1] of the discrete states $P(z_{0:t}|y_{1:t})$. This distribution can be derived from the posterior distribution $P(dx_{0:t}, z_{0:t}|y_{1:t})$ by standard marginalisation. The posterior density satisfies the following recursion

$$\begin{aligned} p(x_{0:t}, z_{0:t}|y_{1:t}) &= p(x_{0:t-1}, z_{0:t-1}|y_{1:t-1}) \\ &\quad \times \frac{p(y_t|x_t,z_t)p(x_t,z_t|x_{t-1},z_{t-1})}{p(y_t|y_{1:t-1})}. \end{aligned} \quad (1)$$

This recursion involves intractable integrals. One, therefore, has to resort to some form of numerical approximation scheme.

In the particle filtering setting, we use a weighted set of samples (particles) $\{(x_{0:t}^{(i)}, z_{0:t}^{(i)}), w_t^{(i)}\}_{i=1}^N$ to approximate the posterior with the following point-mass distribution

$$\widehat{P}_N(dx_{0:t}, z_{0:t}|y_{1:t}) = \sum_{i=1}^{N} w_t^{(i)} \delta_{x_{0:t}^{(i)}, z_{0:t}^{(i)}}(dx_{0:t}, z_{1:t}),$$

where $\delta_{x_{0:t}^{(i)}, z_{0:t}^{(i)}}(dx_{0:t}, z_{1:t})$ denotes the Dirac-delta function. This approximation can be updated recursively using equation (1), as shown in Figure 1. (For simplicity we only show the update of the marginal for x in the figure.)

By considering the following factorisation

$$p(x_{0:t}, z_{0:t}|\, y_{1:t}) = p(x_{0:t}|\, y_{1:t}, z_{0:t})\, p(z_{0:t}|\, y_{1:t}),$$

it is possible to design more efficient algorithms. The density $p(x_{0:t}|\, y_{1:t}, z_{0:t})$ is Gaussian and can be computed analytically if we know the marginal posterior density $p(z_{0:t}|\, y_{1:t})$. This density satisfies the alternative recursion

$$p(z_{0:t}|y_{1:t}) = p(z_{0:t-1}|y_{1:t-1}) \frac{p(y_t|y_{1:t-1}, z_{0:t})\, p(z_t|z_{t-1})}{p(y_t|y_{1:t-1})} \quad (2)$$

[1]NOTATION: For a generic vector θ, we adopt the notation $\theta_{1:t} \triangleq (\theta_1, \theta_2, \ldots, \theta_t)'$ to denote all the entries of this vector at time t. For simplicity, we use θ_t to denote both the random variable and its realisation. Consequently, we express continuous probability distributions using $P(d\theta_t)$ instead of $\Pr(\theta_t \in d\theta_t)$ and discrete distributions using $P(\theta_t)$ instead of $\Pr(\theta_t = \theta_t)$. If these distributions admit densities with respect to an underlying measure μ (counting or Lebesgue), we denote these densities by $p(\theta_t)$. For example, when considering the space $\mathbb{R}^n$, we will use the Lebesgue measure, $\mu = d\theta_t$, so that $P(d\theta_t) = p(\theta_t)\, d\theta_t$.

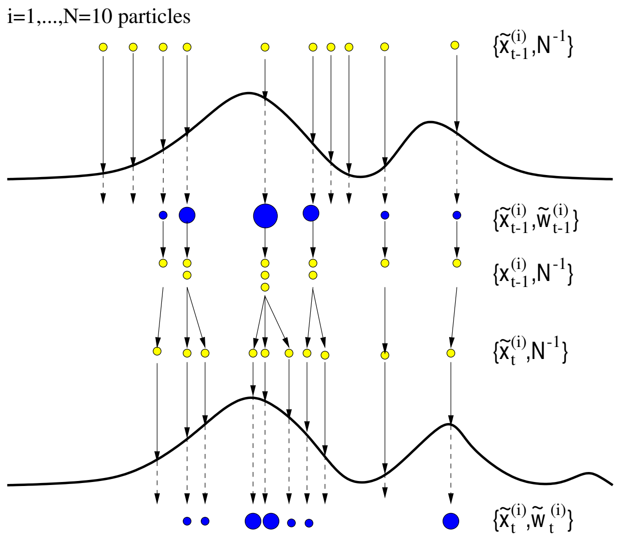

Figure 1. In this example, a particle filter starts at time $t-1$ with an unweighted measure $\{\widehat{x}_{t-1}^{(i)}, N^{-1}\}$, which provides an approximation of $p(x_{t-1}|y_{1:t-2})$. For each particle we compute the importance weights using the information at time $t-1$. This results in the weighted measure $\{\widehat{x}_{t-1}^{(i)}, \widehat{w}_{t-1}^{(i)}\}$, which yields an approximation of $p(x_{t-1}|y_{1:t-1})$. Subsequently, a resampling step selects only the "fittest" particles to obtain the unweighted measure $\{x_{t-1}^{(i)}, N^{-1}\}$. This yields an approximation of $p(x_{t-1}|y_{1:t-1})$ that is "concentrated" on the most likely hypothesis, thereby allowing for nonstationary tracking. Finally, a sampling (prediction) step introduces variety, resulting in the measure $\{\widehat{x}_t^{(i)}, N^{-1}\}$, which is an approximation of $p(x_t|y_{1:t-1})$.

If equation (1) does not admit a closed-form expression, then equation (2) does not admit one either and sampling-based methods are still required. (Also note that the term $p(y_t|y_{1:t-1}, z_{0:t})$ in equation (2) does not simplify to $p(y_t|z_t)$ because there is a dependency on past values through $x_{0:t}$.) Now assuming that we can use a weighted set of samples $\{z_{0:t}^{(i)}, w_t^{(i)}\}_{i=1}^N$ to represent the marginal posterior distribution

$$\widehat{P}_N(z_{0:t}|y_{1:t}) = \sum_{i=1}^{N} w_t^{(i)} \delta_{z_{0:t}^{(i)}}(z_{1:t}),$$

the marginal density of $x_{0:t}$ is a Gaussian mixture

$$\begin{aligned} \widehat{p}_N(x_{0:t}|y_{1:t}) &= \int p(x_{0:t}|z_{0:t}, y_{1:t}) dP(z_{0:t}|y_{1:t}) \\ &= \int p(x_{0:t}|z_{0:t}, y_{1:t}) \sum_{i=1}^{N} w_t^{(i)} \delta_{z_{0:t}^{(i)}}(z_{0:t}) \\ &= \sum_{i=1}^{N} w_t^{(i)} p(x_{0:t}|y_{1:t}, z_{0:t}^{(i)}) \end{aligned}$$

that can be computed efficiently with a stochastic bank of Kalman filters as shown in Section 4. A Rao-Blackwellised filter that combines this marginalisation and sampling of z_t is described in Section 4.

3. PARTICLE FILTERING

Given N particles (samples) $\{x_{0:t-1}^{(i)}, z_{0:t-1}^{(i)}\}_{i=1}^{N}$ at time $t-1$, approximately distributed according to the distribution $P(dx_{0:t-1}^{(i)}, z_{0:t-1}^{(i)}|y_{1:t-1})$, particle filters enable us to compute N particles $\{x_{0:t}^{(i)}, z_{0:t}^{(i)}\}_{i=1}^{N}$ approximately distributed according to the posterior $P(dx_{0:t}^{(i)}, z_{0:t}^{(i)}|y_{1:t})$, at time t. Since we cannot sample from the posterior directly, the PF update is accomplished by introducing an appropriate importance proposal distribution $Q(dx_{0:t}, z_{0:t})$ from which we can obtain samples. The basic algorithm consists of two steps: sequential importance sampling and selection.

<u>*Sequential importance sampling step*</u>

- For $i = 1, ..., N$, sample from the transition priors

$$\begin{aligned} \widehat{z}_t^{(i)} &\sim \Pr(z_t|z_{t-1}^{(i)}) \\ \widehat{x}_t^{(i)} &\sim P(dx_t|x_{t-1}^{(i)}, z_t^{(i)}) \end{aligned}$$

and set

$$\left(\widehat{x}_{0:t}^{(i)}, \widehat{z}_{0:t}^{(i)}\right) \triangleq \left(\widehat{x}_t^{(i)}, \widehat{z}_t^{(i)}, x_{0:t-1}^{(i)}, z_{0:t-1}^{(i)}\right)$$

- For $i = 1, ..., N$, evaluate and normalize the importance weights

$$w_t^{(i)} \propto p\left(y_t|\widehat{x}_t^{(i)}, \widehat{z}_t^{(i)}\right)$$

<u>*Selection step*</u>

- Multiply/Discard particles $\left\{\widehat{x}_{0:t}^{(i)}, \widehat{z}_{0:t}^{(i)}\right\}_{i=1}^{N}$ with respect to high/low importance weights $w_t^{(i)}$ to obtain N particles $\left\{x_{0:t}^{(i)}, z_{0:t}^{(i)}\right\}_{i=1}^{N}$.

Figure 2. Simple sequential Monte Carlo algorithm at time t. For filtering purposes (PF), there is no need for storing or resampling the past trajectories.

Sequential importance sampling step

In generic sequential Monte Carlo (SMC) simulation, one needs to extend the current paths $\{x_{0:t-1}^{(i)}, z_{0:t-1}^{(i)}\}_{i=1}^{N}$ to obtain new paths $\{\widehat{x}_{0:t}^{(i)}, \widehat{z}_{0:t}^{(i)}\}_{i=1}^{N}$ using the proposal distribution $Q\left(d\widehat{x}_{0:t}, \widehat{z}_{0:t}|y_{1:t}\right)$ given by the integral

$$\int Q(d\widehat{x}_{0:t}, \widehat{z}_{0:t}|x_{0:t-1}, z_{0:t-1}, y_{1:t})\, dP(x_{0:t-1}, z_{0:t-1}|y_{1:t-1})$$

To make this integral tractable, we only propose to modify the particles at time t, and leave the past trajectories intact. In other words $Q\left(d\widehat{x}_{0:t}, \widehat{z}_{0:t}|x_{0:t-1}, z_{0:t-1}, y_{1:t}\right)$ is set to

$$Q\left(d\widehat{x}_t, \widehat{z}_t|x_{0:t-1}, z_{0:t-1}, y_{1:t}\right)\delta_{x_{0:t-1}, z_{0:t-1}}\left(d\widehat{x}_{0:t-1}, \widehat{z}_{0:t-1}\right)$$

and consequently $Q\left(d\widehat{x}_{0:t}, \widehat{z}_{0:t}|y_{1:t}\right)$ is equal to

$$P(dx_{0:t-1}, z_{0:t-1}|y_{1:t-1})Q(d\widehat{x}_t, \widehat{z}_t|x_{0:t-1}, z_{0:t-1}, y_{1:t})$$

The samples from $Q(\cdot)$, must be weighted by the importance weights

$$\begin{aligned} w_t &= \frac{P(d\widehat{x}_{0:t}, \widehat{z}_{0:t}|y_{1:t})}{Q(d\widehat{x}_{0:t}, \widehat{z}_{0:t}|y_{1:t})} \\ &= \frac{P(dx_{0:t-1}, z_{0:t-1}|y_{1:t})}{P(dx_{0:t-1}, z_{0:t-1}|y_{1:t-1})} \\ &\quad \times \frac{P(d\widehat{x}_t, \widehat{z}_t|x_{0:t-1}, z_{0:t-1}, y_{1:t})}{Q(d\widehat{x}_t, \widehat{z}_t|x_{0:t-1}, z_{0:t-1}, y_{1:t})} \\ &\propto \frac{p\left(y_t|\widehat{x}_t, \widehat{z}_t\right)p\left(\widehat{x}_t, \widehat{z}_t|x_{0:t-1}, z_{0:t-1}, y_{1:t}\right)}{q_t\left(\widehat{x}_t, \widehat{z}_t|x_{0:t-1}, z_{0:t-1}, y_{1:t}\right)}. \end{aligned} \tag{4}$$

In our case, the transition prior $p\left(\widehat{x}_t, \widehat{z}_t|x_{0:t-1}, z_{0:t-1}, y_{1:t}\right)$ simplifies to the Markov density $p\left(\widehat{x}_t, \widehat{z}_t|x_{t-1}, z_{t-1}\right)=$ $p\left(\widehat{x}_t|x_{t-1}, z_{t-1}\right)p\left(\widehat{z}_t|z_{t-1}\right)$; though this simplification is not required in order to implement SMC algorithms.

From equation (4), we note that the optimal importance distribution is $Q(\cdot|\cdot) = P(d\widehat{x}_t, \widehat{z}_t|x_{0:t-1}, z_{0:t-1}, y_{1:t})$. (When using this proposal, one might still encounter difficulties if the ratio of the first two terms of equation (4) differs significantly from 1 [3], [14].) The optimal importance distribution can be difficult to evaluate. One can adopt, instead, the transition prior as proposal distribution

$$Q(\cdot|\cdot) = P\left(d\widehat{x}_t|x_{t-1}, z_{t-1}\right)\Pr\left(\widehat{z}_t|z_{t-1}\right),$$

in which case the importance weights are given by the likelihood function

$$w_t \propto p\left(y_t|\widehat{x}_t, \widehat{z}_t\right).$$

This is the algorithm shown in Figure 2. It has appeared under many names, including condensation [11], survival of the fittest [12] and the bootstrap filter [10]. The importance sampling framework allows us to design more principled and "clever" proposal distributions. For instance, one can adopt suboptimal filters and other approximation methods that make use of the information available at time t to generate the proposal distribution [8], [6], [14], [16]. In fact, in some restricted situations, one may interpret the likelihood as a distribution in terms of the states and sample from it directly. In doing so, the importance weights become equal to the transition prior [9].

Selection step

A selection scheme associates to each particle $(\widehat{x}_{0:t}^{(i)}, \widehat{z}_{0:t}^{(i)})$ a number of "children", say $N_i \in \mathbb{N}$, such that $\sum_{i=1}^{N} N_i = N$. This selection step is what allows us to track moving target distributions efficiently. There are various selection schemes in the literature, but their performance varies in terms of $var\left[N_i\right]$. We choose a minimum variance sampling algorithm [5], [13]. We sample a set of N points U in the interval $[0, 1]$, each of the points a distance N^{-1} apart. The number of children N_i is taken to be the number of points that lie between $\sum_{j=1}^{i-1} w_t^{(j)}$ and $\sum_{j=1}^{i} w_t^{(j)}$. This strategy introduces a variance on

N_i even smaller than multinomial or residual resampling schemes, namely $var(N_i) = \overline{N}_t w_t'^{(i)} \left(1 - \overline{N}_t w_t'^{(i)}\right)$, where $w_t'^{(i)} = \overline{N}_t^{-1} \left(\widetilde{w}_t^{(i)} N - \widetilde{N}_i\right)$, $\widetilde{N}_i = \left\lfloor N \widetilde{w}_t^{(i)} \right\rfloor$ and $\overline{N}_t = N - \sum_{i=1}^{N} \widetilde{N}_i$. Its computational complexity is $O(N)$.

4. Rao-Blackwellised Particle Filtering

The RBPF is similar to the PF, but we only sample the discrete states. Then for each sample of the discrete states, we update the mean and covariance of the continuous states using exact computations. In particular, we sample $z_t^{(i)}$ and then propagate the mean $\mu_t^{(i)}$ and covariance $\Sigma_t^{(i)}$ of x_t with a Kalman filter as follows

$$
\begin{aligned}
\mu_{t|t-1}^{(i)} &= A(z_t^{(i)})\mu_{t-1|t-1}^{(i)} + F(z_t^{(i)})u_t \\
\Sigma_{t|t-1}^{(i)} &= A(z_t^{(i)})\Sigma_{t-1|t-1}^{(i)} A(z_t^{(i)})^{\mathsf{T}} + B(z_t^{(i)})B(z_t^{(i)})^{\mathsf{T}} \\
S_t^{(i)} &= C(z_t^{(i)})\Sigma_{t|t-1}^{(i)} C(z_t^{(i)})^{\mathsf{T}} + D(z_t^{(i)})D(z_t^{(i)})^{\mathsf{T}} \\
y_{t|t-1}^{(i)} &= C(z_t^{(i)})\mu_{t|t-1}^{(i)} + G(z_t^{(i)})u_t \\
\mu_{t|t}^{(i)} &= \mu_{t|t-1}^{(i)} + \Sigma_{t|t-1}^{(i)} C(z_t^{(i)})^{\mathsf{T}} S_t^{-1(i)} (y_t - y_{t|t-1}^{(i)}) \\
\Sigma_{t|t}^{(i)} &= \Sigma_{t|t-1}^{(i)} - \Sigma_{t|t-1}^{(i)} C(z_t^{(i)})^{\mathsf{T}} S_t^{-1(i)} C(z_t^{(i)})\Sigma_{t|t-1}^{(i)}
\end{aligned}
\quad (5)
$$

where $\mu_{t|t-1} \triangleq \mathbb{E}(x_t | y_{1:t-1})$, $\mu_{t|t} \triangleq \mathbb{E}(x_t | y_{1:t})$, $y_{t|t-1} \triangleq \mathbb{E}(y_t | y_{1:t-1})$, $\Sigma_{t|t-1} \triangleq cov(x_t | y_{1:t-1})$, $\Sigma_{t|t} \triangleq cov(x_t | y_{1:t})$ and $S_t \triangleq cov(y_t | y_{1:t-1})$. Hence, using the prior proposal for z_t and applying equation (2), we find that the importance weights for z_t are given by the predictive density

$$p(y_t | y_{1:t-1}, z_{1:t}) = \mathcal{N}\left(y_t; y_{t|t-1}, S_t\right). \quad (6)$$

The new algorithm is shown in Figure 3.

5. Experiments

In our experiments, we use the state-space model to generate synthetic data and then compare the performance of the PF and RBPF algorithms. We set the matrices A and C at random, $B = D = 0.01I$, $F = G = 0$, $\mu_0 = 0$ and $\Sigma_0 = 0.1I$. In the first experiment, we set $n_x = 1$, $n_y = n_z = 2$, $P(z_0) = (0.5, 0.5)$ and

$$P(z_t|z_{t-1}) = \begin{bmatrix} 0.1 & 0.9 \\ 0.1 & 0.9 \end{bmatrix}.$$

Here, the first state is visited only on rare occasions. We repeated the experiment 10 times for 50, 100, 500 and 1000 particles. The results are shown in Figure 4. Clearly, the RBPF outperforms the PF algorithm significantly. Moreover, it only seems to require a few particles. Figure 5 shows one typical run with parameters $N = 100$, $n_z = 3$, $n_x = n_y = 2$, $P(z_0) = (0.5, 0.5, 0.5)$ and

$$P(z_t|z_{t-1}) = \begin{bmatrix} 0.1 & 0.5 & 0.4 \\ 0.1 & 0.6 & 0.3 \\ 0.1 & 0.3 & 0.6 \end{bmatrix}.$$

Figure 6 depicts the propagated discrete distribution of z.

Sequential importance sampling step

- For $i = 1, ..., N$, set $\widehat{\mu}_{t|t-1}^{(i)} \triangleq \mu_{t|t-1}^{(i)}$, $\widehat{\Sigma}_{t|t-1}^{(i)} \triangleq \Sigma_{t|t-1}^{(i)}$, sample

$$\widehat{z}_t^{(i)} \sim \Pr(z_t | z_{t-1}^{(i)})$$

- For $i = 1, ..., N$, evaluate and normalize the importance weights

$$w_t^{(i)} \propto p\left(y_t | y_{1:t-1} \widehat{z}_t^{(i)}\right)$$

Selection step

- Multiply/Discard particles $\left\{\widehat{\mu}_{t|t-1}^{(i)}, \widehat{\Sigma}_{t|t-1}^{(i)}, \widehat{z}_t^{(i)}\right\}_{i=1}^{N}$ with respect to high/low importance weights $w_t^{(i)}$ to obtain N particles $\left\{\mu_{t|t-1}^{(i)}, \Sigma_{t|t-1}^{(i)}, z_t^{(i)}\right\}_{i=1}^{N}$.

Updating step

- For $i = 1, ..., N$, use one step of the Kalman recursion to compute the minimum statistics $\left\{\mu_{t+1|t}^{(i)}, \Sigma_{t+1|t}^{(i)}, y_{t+1|t}^{(i)}, S_{t+1}^{(i)},\right\}$ given $\left\{z_t^{(i)}, \mu_{t|t-1}^{(i)}, \Sigma_{t|t-1}^{(i)}\right\}$.

Figure 3. RBPF algorithm at time t.

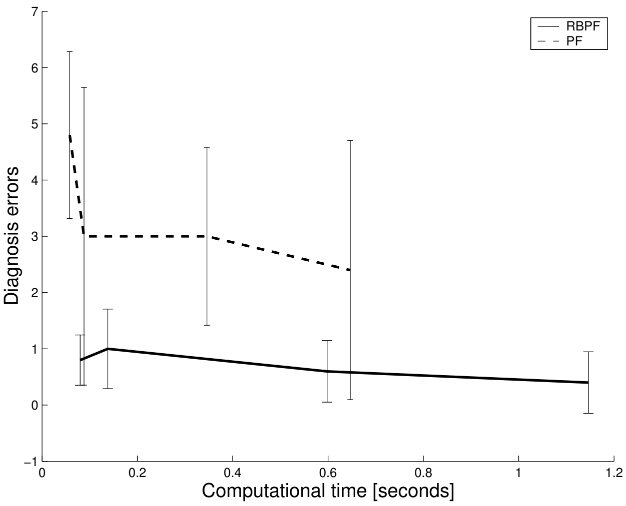

Figure 4. Results for 50, 100, 500 and 1000 particles. The RBPF outperforms the PF significantly.

6. Conclusions

We proposed the use of Rao-Blackwellisation to improve the state-of-the-art algorithms for state estimation and fault diagnosis. The results show a considerable improvement.

We are currently addressing the question of diagnosing very rare events. This question is of fundamental importance in the diagnosis of faults in planetary rovers. To solve this problem, a recent paper [17] introduced a particle filter that penalises particles outside the region of error. This approach can be easily combined with our Rao-

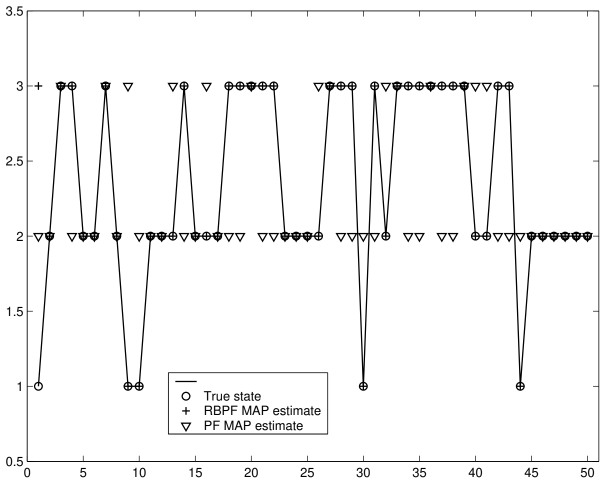

Figure 5. Typical run. The RBPF makes less estimation mistakes.

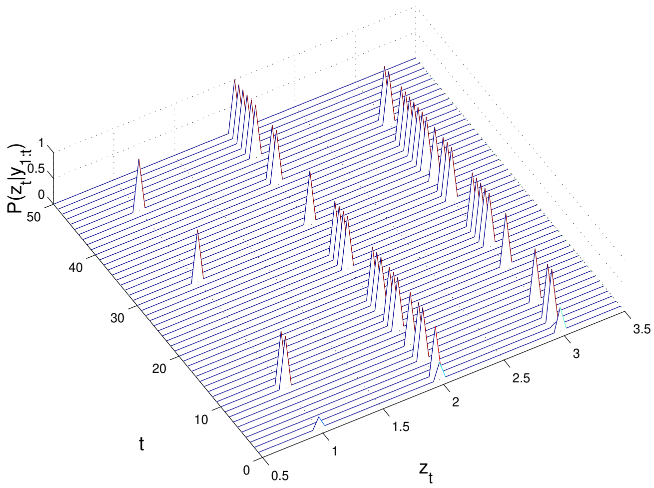

Figure 6. Distribution of the discrete states computed with the RBPF.

Blackwellisation scheme. However, this is not enough. Instead of penalising particles outside the region of error, we must bias the importance proposal distribution to generate more samples in the region of error. This is essential because the region of error is exponentially small with respect to the size of the state space. Fortunately, there are some guidelines for solving this problem in the communications literature concerning importance sampling for bit error detection [15]. Here, adaptive techniques or large deviation changes of measure can be used to generate appropriate proposal distributions. We are currently investigating ways of extending this batch framework to our on-line framework.

7. Acknowledgments

I would like to thank Arnaud Doucet, Deva Ramanan, Vandi Verma and Richard Washington.

References

[1] H Akashi and H Kumamoto. Random sampling approach to state estimation in switching environments. *Automatica*, 13:429–434, 1977.

[2] C Andrieu, N de Freitas, and A Doucet. Sequential Bayesian estimation and model selection applied to neural networks. Technical Report CUED/F-INFENG/TR 341, Cambridge University Engineering Department, May 1999.

[3] C Andrieu, A Doucet, and E Punskaya. Sequential Monte Carlo methods for optimal filtering. In A Doucet, N de Freitas, and N J Gordon, editors, *Sequential Monte Carlo Methods in Practice*. Springer-Verlag, 2001.

[4] G Casella and C P Robert. Rao-Blackwellisation of sampling schemes. *Biometrika*, 83(1):81–94, 1996.

[5] D Crisan. Particle filters - a theoretical perspective. In A Doucet, N de Freitas, and N J Gordon, editors, *Sequential Monte Carlo Methods in Practice*. Springer-Verlag, 2001.

[6] N de Freitas, M Niranjan, A H Gee, and A Doucet. Sequential Monte Carlo methods to train neural network models. *Neural Computation*, 12(4):955–993, 2000.

[7] A Doucet, N de Freitas, and N J Gordon, editors. *Sequential Monte Carlo Methods in Practice*. Springer-Verlag, 2001.

[8] A Doucet, S Godsill, and C Andrieu. On sequential Monte Carlo sampling methods for Bayesian filtering. *Statistics and Computing*, 10(3):197–208, 2000.

[9] D Fox, S Thrun, W Burgard, and F Dellaert. Particle filters for mobile robot localization. In A Doucet, N de Freitas, and N J Gordon, editors, *Sequential Monte Carlo Methods in Practice*. Springer-Verlag, 2001.

[10] N J Gordon, D J Salmond, and A F M Smith. Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *IEE Proceedings-F*, 140(2):107–113, April 1993.

[11] M Isard and A Blake. Contour tracking by stochastic propagation of conditional density. In *European Conference on Computer Vision*, pages 343–356, Cambridge, UK, 1996.

[12] K Kanazawa, D Koller, and S Russell. Stochastic simulation algorithms for dynamic probabilistic networks. In *Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence*, pages 346–351, Morgan Kaufmann, 1995.

[13] G Kitagawa. Monte Carlo filter and smoother for non-Gaussian nonlinear state space models. *Journal of Computational and Graphical Statistics*, 5:1–25, 1996.

[14] M K Pitt and N Shephard. Filtering via simulation: Auxiliary particle filters. *Journal of the American Statistical Association*, 94(446):590–599, 1999.

[15] P J Smith, M Shafi, and H Gao. Quick simulation: A review of importance sampling techniques in communications systems. *IEEE Journal on Selected Areas in Communications*, 15(4):597–613, 1997.

[16] R van der Merwe, A Doucet, N de Freitas, and E Wan. The unscented particle filter. Technical Report CUED/F-INFENG/TR 380, Cambridge University Engineering Department, June 2000.

[17] V Verma, J Langford, and R Simmons. Non-parametric fault identification for space rovers. In *International Symposium on Artificial Intelligence and Robotics in Space (iSAIRAS)*, June 2001.

[18] R Washington. On-board real-time state and fault identification for rovers. In *IEEE International Conference on Robotics and Automation*, 2000.

Nando de Freitas *is an assistant professor in the Department of Computer Science at the University of British Columbia. His research interests include computational statistics, machine learning, robotics, multimedia and Monte Carlo methods. Dr. de Freitas obtained B.Sc. and M.Sc. degrees in electrical engineering from Wits University in 1994 and 1996, and a Ph.D. in information engineering from Cambridge University in 1999. From 1999 to 2001, he worked as a post-doctoral fellow in the computer science division at UC Berkeley.*